석 사 학 위 논 문

Master's Thesis

이미지 분류를 위한 베이지안 네트워크

Bayesian Network for Image Classification

순 밍 양 (孫 銘 陽 Sun, Mingyang)

전산학과

Department of Computer Science

KAIST

2015

이미지 분류를 위한 베이지안 네트워크

Bayesian Network for Image Classification

Bayesian Network for Image Classification

Advisor : Professor Sung-Eui Yoon

by

Sun, Mingyang
Department of Computer Science
KAIST

A thesis submitted to the faculty of the KAIST in partial fulfillment of the requirements for the degree of Master of Science in Engineering in the Department of Computer Science

Daejeon, Korea
2014. 11. 28.
Approved by

Professor Sung-Eui Yoon
Advisor

이미지 분류를 위한 베이지안 네트워크

순 밍 양

위 논문은 한국과학기술원 석사학위논문으로 학위논문심사 위원회에서 심사 통과하였음.

2014년 12월 15일

심사위원장 윤 성 의 (인)

심사위원 백 종 문 (인)

심사위원 현 준 수 (인)

MCS 20124326 순 밍 양. Sun, Mingyang. Bayesian Network for Image Classification. 이미지 분류를 위한 베이지안 네트워크. Department of Computer Science . 2015. 21p. Advisor Prof. Sung-Eui Yoon. Text in English.

Abstract

Under the conditional independence assumption among local features , naive bayes nearest neighbor classifier (NBNN) computes the direct classification results without any learning or training phase. It outperforms all the previous works in non-parametric classifiers. However, among the local features, there are strong dependences. This assumption, which is obviously against the compositionality of objects, weaken the performance of NBNN. Therefore, we propose a novel Bayesian Network for image classification, which consider the dependence among the local features. In order to utilize the dependences to improve the classification results, we further define the relationship between the high-level and low-level features, by which we optimize the Bayesian Network with relations between high-level and low-level features. By testing our method against previous works in the dataset of Caltech101, our optimized method achieves up to 20% relative accuracy improvement over prior methods in the similar time consuming with NBNN (less than 0.001s difference).

Contents

List of Tables

List of Figures

1. Introduction

Image classification has been a major research direction in computer vision. Its goal is to assign the query image to the belonged class by classifier. By far, the classifiers can be roughly divided into two families: learning-based (parametric) method [10, 4, 3], such as SVM[14], boosting[21], etc., which requires an intensive learning/training phase for classifier parameters to classify the image; and non-parametric methods (no learning phase) [1, 2, 5, 15, 11, 6, 8, 9, 7], which doesn't need any learning/training phase. Meanwhile, non-parameter can easily work with a huge number of classes but learning-based method cannot.

In [2], Boiman et al. proposed a novel, efficient and non-parametric method for image classification, the Naïve Bayes Nearest Neighbor (NBNN) classifier. Given a set of local image descriptors extracted from one query image, instead of quantizing the descriptors to compute the likelihood/distance of image-to-image, NBNN provides the extremely simple scheme to directly compute the distance of image-to-class without learning/training phases. In spite of the absence of the simplicity and the absence of learning/training phases in NBNN, it still achieves surprisingly remarkable performance in image classification, which ranks among the top learning-based image classifiers. It is mainly attributed to 1) the lack of descriptors quantization (e.g.: bag of words), and 2) the use of "image-to-class" distance instead of computing the "image-to-image" distance (e.g.: vocabulary tree).

However, NBNN still suffers its performance by the following defects: I) computation complexity during testing is so high; II) the original NBNN only can work well in the balanced datasets; III) the assumption of the independency of each descriptor violates the performance of the NBNN classifiers; IV) even though NBNN performs remarkably in image classification. Performance of image classification for large scale image retrieval, some optimized methods (optimized NBNN [1], Class-to-Image distance [7], the NBNN kernel [6], local NBNN [5]) are proposed to address either of the issues stated above. However, no work has been done to tackle the problem of conditional independence assumption which is the most harmful to the NBNN classification.

Considering the local descriptors, the dependences among the local features are shown in Fig.1, where one object (camera) is constructed by parts and one part is constructed by

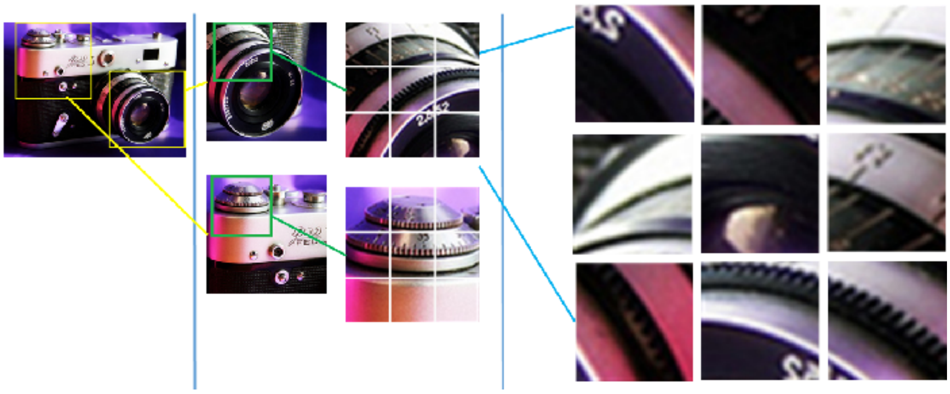

Figure 1.1: The camera is constructed by several parts (high-level features), and every part is constructed by several features (low-level features).

several features. This is so-called compositionality of objects. The locally near features can be together to represent one part of the object and several parts can be together to represent the object. We found that the compositionality of objects is layered and deep. And by considering the conditional dependence of parameters, Naïve Bayes can be transformed to Bayesian Network, that Naïve Bayes is the simplest form of Bayesian Network. Additionally, extracting the optimal dependence among local descriptors for Bayesian Network is an NP problem. Instead, we model the dependences based on the layered structure of the compositionality of objects.

Therefore, in this paper, we will discuss the dependences among local descriptors, introduce the layered structure of local descriptors, and model the Bayesian Network to tackle the weak assumption of the conditional independence among descriptors to tackle the weak assumption of the conditional independence among features. We keep the discriminative power of descriptors by no learning/training phase. And it is sufficient and easy to work with large datasets.

2. Related Work

In this section we review prior techniques on NBNN classifier [2, 5], local features [12, 18], and Bayesian Network [13, 16, 19, 20].

2.1 Naive Bayes Nearest Neighbor (NBNN) Classifier

The original NBNN [2, 5, 6] as shown in Fig. 2.1is so simple and efficient in providing the classification results. Given a query image Q, the original NBNN assigns Q to one class based on the Maximum Likelihood (ML) Estimation under the uniform prior among a possible class set, C:

$$\hat{C} = argmax_c P(C|Q) \tag{2.1}$$

Assuming a uniform prior over all classes and applying Bayes' rule, Eq. 2.1 can be transformed to:

$$\hat{C} = argmax_c log(P(Q|C)) \tag{2.2}$$

Let $D^q = (d_1, d_2, \ldots, d_n)$ denotes all the descriptors of query image Q, which are assumed to be conditionally independent. Therefore, the ML estimation for NBNN can be transformed as follows:

$$\hat{C} = argmax_c log[\Pi_{i=1}^n P(d_i|C)] = argmax_c[\Sigma_{i=1}^n log P(d_i|C)] \tag{2.3}$$

And NBNN takes the Parzen Kernel estimator to compute the posterior probability for classifying the images. As a result, the classifier is approximated as the following:

$$\hat{C} = argmin_c \Sigma_{d_i \in D^q} \|d_i - NN_c(d_i)\|^2 \tag{2.4}$$

where $NN_c(d_i)$ is the nearest neighbor feature of d_i among features extracted from the class C.

In order to optimize the NBNN classifier to work for the large scale image retrieval, some works were proposed to address the problems stated above.[1, 6, 5, 8, 9, 7] But neither of those works has ever tried to tackle the problem of conditional independence assumption of NBNN. Some works focus on the problem of expensive time consuming [5], some on the

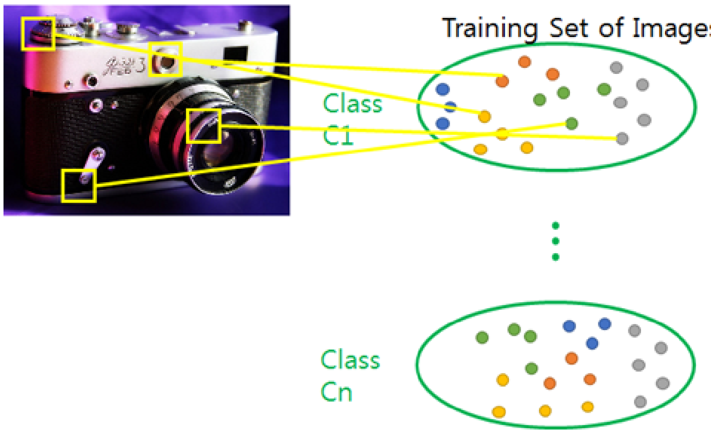

Figure 2.1: Naive Bayes Nearest Neighbor Classifier: given a query image, it searchs for the nearest neighbor of each descriptor extracted from the query image and sum up all the distances of descriptors to their nearest neighbors.

problem of the unbalanced datasets [6], and some provide the capable scheme to retrieve images for the query image [1, 7]. By applying some methods stated above, we can achieve the better results over NBNN with the independence assumption.

In this paper, we take an eye on the problem caused by the conditional independence assumption for image classification. Previously, Naïve Bayes Nearest Neighbor classification worked well in the text classification under the assumption that all the words in one document are independent. This assumption is obviously wrong in the real world. It is well known that the words or texts in one document work together for a mutual concept. Therefore a lot of excellent researchers have focused on this field in the latest 10 years to improve the performance of classification results. Inspired by these works, we try to employ the more empirical Bayesian classifier instead of the Naïve Bayes classifier to provide the more accurate and efficient NBNN classifier for image classification.

2.2 Local Features and Dependences among Local Features

A set of local features are extracted or computed to represent one image. With the rapid development in computer vision, a lot of algorithms have been studied to describe the local features. Scale-invariant feature transform (SIFT), for instance, is an algorithm proposed by David Lowe [12] to detect or describe local features. Lowe's method to detect the key points in one image actually omits some discriminative feature of one object or image, because some discriminative feature does not show in the key point. For example, in Fig.2.2, the feature d_1 is a discriminative feature of a camera compared with the other objects but it doesn't locate in the key point. Therefore, dense SIFT [18], which has no scale and location selection, can make sure the discriminative power of features. It is produced on a regular grid (location) using constant in the patch size (scale).

In Fig.2.2, local features of $d_1, d_2, \ldots, d_9$ are extracted from the part D of the camera described by dense SIFT. And d_1, d_2, d_3, d_4 are nearest neighbors, so that they can be together to represent the feature of $D^{'}$. Therefore, given the camera class, the combination of d_1 and d_2 has the higher probability to decide the image than the combination ofd_1 and d_9, and also higher than d_1 or d_2. It is because of the dependences among the local features.

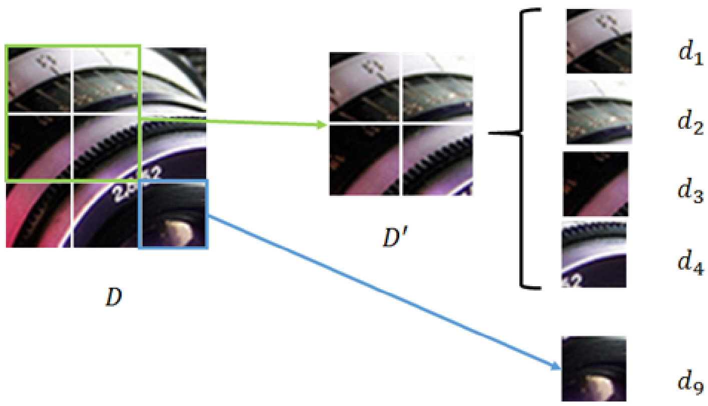

Figure 2.2: The camera is constructed by several parts (high-level features), and every part is constructed by several features (low-level features).

As stated above, the dependences among the near neighbor local features exist and the combination of local features can help us to classify the images. It also shows that the assumption of conditional independence among all the local features weaken the performance of NBNN classifier a lot. In our work, we build the layered structure of features with different scaled descriptors and utilize the locally dependences to tackle the independence assumption and gain the better classification results.

2.3 Bayesian Network

Naïve Bayes is the simplest form of Bayesian Network [19, 17, 13, 20] which only has two layers: one parent node and other nodes. Its simple structure is relayed on the assumption that given the parent node the other nodes are independent, which is called conditional independence. In order to tackle the conditional independence assumption in NBNN, we have to build the Bayesian Network to computer the probability.

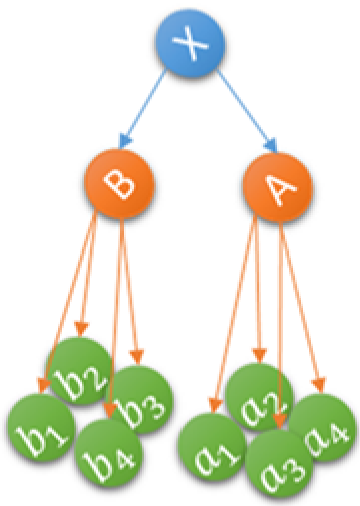

Figure 2.3: The graph is modelled by three layers of nodes, in which node X is the parent node of node A and B, and A and B have four separarent children nodes.

And in Bayesian Network, there is a relative simple algorithm for determining whether two variables in a Bayesian network are conditional independent in the layered structure. For example, in Fig.2.3, the probability can be computed as follows:

$$P(X, A, B, a_1, \ldots, b_4) = P(X)P(A, B|X)P(a_1, \ldots, a_4|A)P(b_1, \ldots, b_4|B) \tag{2.5}$$

where A and B are conditional independent under parent node $X, a_1, \ldots, a_4$are conditional independent under their parent node A, and $b_1, \ldots, b_4$ are conditional independent under their parent node B. And for $X, a_1, \ldots, a_4$ and $b_1, \ldots, b_4, A$ and B are called d-separation, under which $X, a_1, \ldots, a_4$ and $b_1, \ldots, b_4$ are conditional independent. Therefore, d-separation [19, 17, 13, 20] is used to model the dependences among the nodes. In our work, we build the features' structure with the d-separation nodes like Fig.2.3.

3. Bayesian Network for Image Classification

In order to tackle the weak assumption of conditional independence in NBNN, we need to find out the dependences among local features for Bayesian Network. However, searching for the optimal dependences in the image is an NP problem. Instead, we model the dependences among local features with d-separation nodes. In this section, we introduce how to build the Bayesian network in local features, how to utilize the Bayesian network structure to calculate the classification results, and the optimization of our idea separately as follows.

3.1 Model Bayesian Network in Local Features

As stated above, given an image, the descriptors can be computed in different scales (patch-size) and different locations (grid). Therefore, for one part of the object, we can represent it in 1 dense SIFT descriptor or 4 dense SIFT descriptors or more descriptors as shown in Fig.3.1

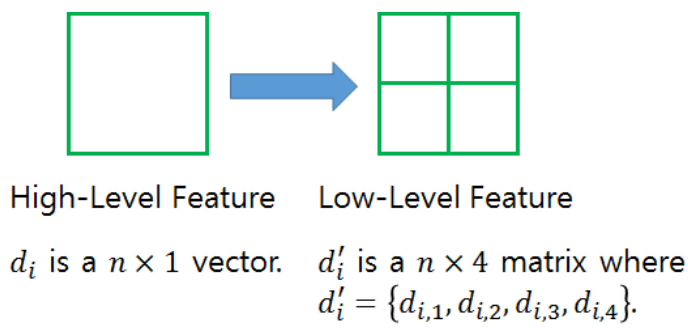

Figure 3.1: For the same patch of feature, it be represented by the same type of descriptors in different scales. In this figure, both d_i and $d_i\,'$ represent the same patch of feature with dense SIFT descriptors. $d_i\,'$ consists of 4 descriptors in the smaller scale compared with d_i.

As shown in Fig.3.1, we call the feature with big scale as high-level feature, equally the feature with smaller scale as low-level feature. It is corresponding to the compositionality of objects, several high-level features can represent one object and several low-level features can represent the high-level feature. And we found that four low-level features are mostly correlated because they are next to each other. In our work we extract one high-level feature with four low-level features in the same patch.

Therefore, we model the bayesian network of local feature in two layers. The first layer is constructed by the high-level features, and name them as parent nodes for every four secific low-level features. The second layer consists of a set of low-level features, which are named as children nodes.

Instead of extracting the local features in one scale, we extract the features in different scales and consider the local features and their relation in layers. The layered structure make the Bayesian network graph, that we can utilize this structure of features for non-parametric classifiers.

Furthermore, for more detailed feature, we can build the bayesian network of the local features in more layers. For instance, we can build 3-layered structure with high-level features, mid-level features, and low-level features, in which one high-level feature as parent node to 4 mid-level features, and one mid-level feature as parent node to 4 low-level features. In this paper, we only introduce the simplest form of the Bayesian network with the 2-layered local features by the descriptor of dense SIFT.

3.2 Bayesian Network for Image Classification

In this section, we explain how to utilize the Bayesian network to classify the images with the layered structure of local features. Becasues there is no learning/training phase in non-parametric classifier, we just construct the training set by several labeled images like NBNN. Different from NBNN, shown in Fig. 3.2, the training set is also layered. And as stated above, given a query image Q, NBNN assigns Q to one class based on the Maximum Likelihood (ML) Estimation under the uniform prior among a possible class set, C:

$$\hat{C} = argmax_c P(C|Q) \tag{3.1}$$

Assuming a uniform prior over all classes and applying Bayes' rule, Eq. 3.1 can be transformed as follows:

$$\hat{C} = argmax_c log[P(Q|C)] \tag{3.2}$$

where C represent the class of training set.

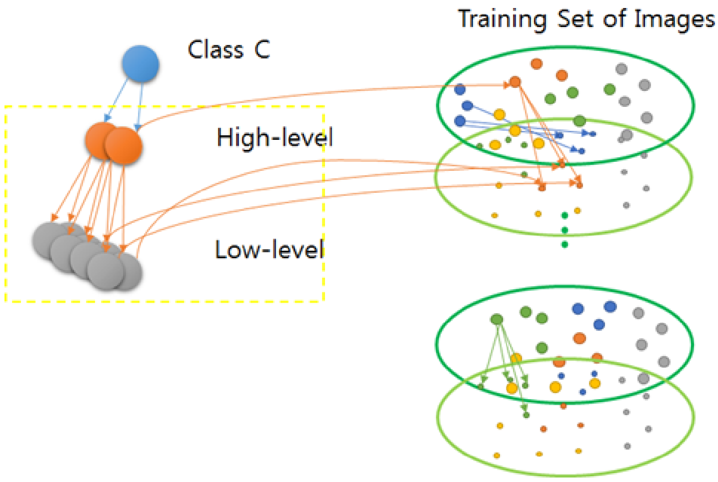

Figure 3.2: Bayesian network for image classification

In Fig. 3.2, the Bayesian Network which consists of the class node, high-level and low-level nodes shows the conditional independence among local features. That is under the parent nodes children nodes are independent. Therefore, Eq. 3.2 can be transformed as follows:

$$\hat{C} = argmax_c log[P(D_h|C)P(D_l|D_h)] \tag{3.3}$$

where D_h represents the high-level features (descriptors), D_l represents D_h's low-level features and each descriptor d_i in D_h has its D_l whcih consists of 4 children nodes $d_{i,1}, d_{i,2}, d_{i,3}, d_{i,4}$. And under the parent node d_i (high-level feature), chilren nodes $d_{i,1}, d_{i,2}, d_{i,3}, d_{i,4}$ (low-level feautures) are independent.(see Eq. 3.4)

$$P(d_{i,1}, \ldots, d_{i,4}|d_i) = \Pi_{j=1}^{m} P(d_{i,j}|d_i) \tag{3.4}$$

In Eq. 3.4, m means how many children nodes one parent node has. In this paper, we set m as 4. And n means how many high features in the image.

Additionally, if this image or object belongs to this class, under this class the high-level features are independent. Therefore, we can get the classification as follows from Eq. 3.3:

$$\hat{C} = argmax_c log[\Pi_{i=1}^{n} P(d_i|C)(\Pi_{j=1}^{m} P(d_{i,j}|d_i))] \tag{3.5}$$

Therefore, from Eq.3.5 under one class, the likehood estimation can be transformed as follows:

$$\hat{C} = argmax_c[\Sigma_{i=1}^{n}(logP(d_i|C) + \Sigma_{j=1}^{m} logP(d_{i,j}|d_i))] \quad (3.6)$$

By a parzen window estimator, with kernel K, the conditional probability of descriptor d_i under class C gives as follows:

$$\hat{P}(d_i|C) = \frac{1}{L}\Sigma_{t=1}^{L} K(d_i - d_t^C) \quad (3.7)$$

Where there are L descriptors in the training set for class C and $d_t^C$is the tth nearest descriptor in class C. In NBNN, t is taken to the extreme by using only the nearest neighbor $NN_c d_i$:

$$\hat{P}(d_i|C) = \frac{1}{L} K(d_i - NN_c(d_i)) \quad (3.8)$$

And the conditional probability of low-level descriptor $d_{i,j}$ under its parent node descriptor d_i gives as follows:

$$\hat{P}(d_{i,j}|d_i) = \frac{1}{l} K(d_{i,j} - NN_{d_i}(d_i, j)) \quad (3.9)$$

where l means how many children nodes do descriptor d_i have, in this paper we set it as 4. Since we estimate the maximum likehood of the high-level descriptor under the class C, we get the nearest neighbor of high-level descriptor d_i, which is $NN_c(d_i)$. In the training set, $d_t = NN_c(d_i)$ has l children nodes $d_{t,1}, \ldots, d_{t,4}$. Therefore, $NN_{d_i}(d_i, j)$ means the nearest neighbor of the low-level feature $d_{i,j}$ among low-level features $d_{t,1}, \ldots, d_{t,4}$.

Since the kernel K is chosen as a Gaussian Kernel, whcih subtituded into Eq.3.8 and Eq.3.9. The classification fuction can be transformed as Eq.3.10 and Eq.3.11:

$$\hat{C} = argmax_c[\Sigma_{i=1}^{n}(log\frac{1}{L}e^{-\frac{1}{2\alpha^2}\|d_i - NN_c(d_i)\|^2} + \Sigma_{j=1}^{m} log\frac{1}{l}e^{-\frac{1}{2\alpha^2}\|d_{i,j} - NN_{d_i}(d_{i,j})\|^2})] \quad (3.10)$$

$$\hat{C} = argmin_c[\Sigma_{i=1}^{n}(\|d_i - NN_c(d_i)\|^2 + \Sigma_{j=1}^{m}\|d_{i,j} - NN_{d_i}(d_{i,j})\|^2)] \quad (3.11)$$

In conclusion, Bayesian Network for image classification can be summarized as follows:

Bayesian Network for Image Classification

1. Compute layered descriptors of high-level descriptors $d_1, \ldots, d_n$ and their low-level descriptors $d_{1,1}, \ldots, d_{1,m}, \ldots, d_{n,m}$

2. Compute the nearear neighbor (NN) of the descriptor

   (a) $\forall d_i$ $\forall C$ compute the NN of d_i in C: $NN_c(d_i)$

   (b) for each $d_{i,j}$ in d_i compute the NN of $d_{i,j}$ in the low-level descriptors of $NN_c(d_i)$: $NN_{d_i}(d_{i,j})$

3. $\hat{C} = argmin_c[\Sigma_{i=1}^{n}(\|d_i - NN_c(d_i)\|^2 + \Sigma_{j=1}^{m}\|d_{i,j} - NN_{d_i}(d_{i,j})\|^2)]$

However, in this algorithm, time complexity to search the nearest neighbor in the layered networks (high-level and low-level features) is much bigger than NBNN for multiple kd-trees. And by this structure, we cannot compute the dependence among high-level and low-level features. Additionally, the low-level features have location information, so that the sequence of the low-level features also represents some information about the relation with the high-level features. For example, d_i has four children nodes $d_{i,1}, \ldots, d_{i,4}$, and both of them represent the same patch of feature, so there is the relation or dependence among the high-level and low-level features. And only the sequence of $d_{i,1}, \ldots, d_{i,4}$ and d_i represent the same feature. This relation can help us to simplify and speed up the algorithm. In the next section, we introduce how to optimize this algorithm with the relation between high-level features and low-level features.

3.3 Optimization

As stated above, for the same patch of feature, it can be represented by d_i or $d_{i,1}, \ldots, d_{i,4}$. And the sequence of $d_{i,1}, \ldots, d_{i,4}$ shows the comparative locations of low-level features to high-level features.Therefore, we define the relation to represent this dependence or correlation between high-level and low-level features as follows.

Relation R For one patch of features, descriptor of high-level fearure D is a $n \times 1$ vector, m descriptors of low-level features $D^{'} = d_1, \ldots, d_m$ is a $n \times m$ matrix. The relation between D and $D^{'}$ is R, where $R = D^{'T} \times D$ which is a $m \times 1$ vector.

In our work, we extract the dense SIFT descriptors in each patch. So, d_i is a 128×1 vector. $d_i^{'} = d_{i,1}, \ldots, d_{i,4}$ is a 128×4 matrix. And the relation r_i between d_i and $d_i^{'} = d_{i,1}, \ldots, d_{i,4}$ is given as follows:

$$r_i = d_i^{'T} \times d_i \tag{3.12}$$

So the low-level features $d_i^{'} = d_{i,1}, \ldots, d_{i,4}$ can be represented by the relation r_i and the high-level feature d_i as shown in Eq.3.13.

$$d_i^{'} = d_i \times r_i^T \tag{3.13}$$

Therefore, the classification fuction Eq.3.11 can be transformed as follows:

$$\hat{C} = argmin_c[\Sigma_{i=1}^{n}(\|d_i - NN_c(d_i)\|^2 + \Sigma_{j=1}^{m}\|d_i \times r_i^T - NN_{d_i}(d_i \times r_i^T)\|^2)] \tag{3.14}$$

where $NN_{d_i}(d_i \times r_i^T)$ means the product of the nearest neighbor d_t of d_i in class C and the relation between d_t and its low-level features $d_t^{'}$, which describe the low-level features.

And d_t is the nearest neighbor of d_i in class C. They are similar. So we can approximate the maximum likehood of the query image to the class C as follows:

$$\hat{C} \approx argmin_c[\Sigma_{i=1}^{n}(\|d_i - NN_c(d_i)\|^2 + \lambda\|r_i - NN_{d_i}(r_i)\|^2)] \tag{3.15}$$

As as shown in Fig. 3.3, our optimized method is quite simple. Instead of store the both high-level and low-level features, we compute the relation between them and construct the training set solely with high-level features and the relations, which saves a lot of space and speeds up a lot. Meanwhile, the relation shows more information of dependence among high-level and low-level features, so that it performs better than the algorithm shown in the last section.

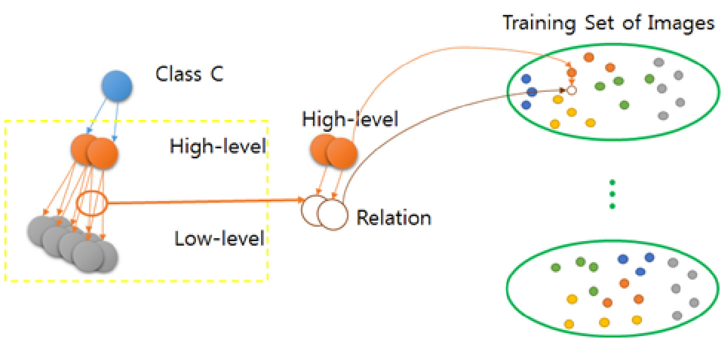

Figure 3.3: Bayesian Network for image classification with the relation between high-level and low-level features

Optimized Bayesian Network for Image Classification

1. Compute layered descriptors of high-level descriptors $d_1, \ldots, d_n$ and their relations with low-level descriptors $r_1, \ldots, r_n$

2. Compute the nearear neighbor (NN) of the descriptor

   (a) $\forall d_i$ $\forall C$ compute the NN of d_i in C: $NN_c(d_i)$

   (b) get the value of the relation $r_i^{'}$ of $NN_c(d_i)$ in C

3. $\hat{C} = argmin_c[\Sigma_{i=1}^{n}(\|d_i - NN_c(d_i)\|^2 + \lambda\|r_i - r_i^{'}\|^2)]$

4. Estimation and Evaluation

We conducted a series of tests on the Intel Quadcore i7 3.60GHz with 16GB memory. We use the FLANN [23] library to compute the fast approximate nearest neighbors, which utilizes multiple and randomized kd-trees. In our experiments we built the indexes of kd-trees in the training set previously. So that we can load the index and search for the nearest neighbors in the fastest way. Compared with FLANN search for nearest neighbors in the training set, FLANN search in the pre-built indexes is tens of times faster, which speeds a lot in the classifcation.

Benchmark We use the Caltech101 benchmark for various tests. This dataset contains 101 categories and about 40 to 800 images per category. By following the experiment protocol of the original NBNN, we randomly choose 15 training images per category as training data for all the tests. For query images we randomly choose 20 test images from the benchmark except for already chosen training images. We have iterated this process two times, and have measured the average precision for each class for accuracy comparisons.

Descriptor For each image we extract densely sampled SIFT descriptors [18] of high-level features at every 16 pixels. The patch size of high-level feature is 32×32 pixels, and the low-level features' is 16×16 pixels Specifically, if the size of an image is smaller than 200, we resize it to 200. Also, when the size is bigger than 450, we resize to 450. In these configurations an image contains 300 to 2000 SIFT descriptors of high-level features.

4.1 Experimental Results

To show benefits of our method, we compare different types of our method against the other state-to-art NBNN methods, as the following:

- NBNN. The original NBNN method.[2]
- Local NBNN. A recently improved NBNN method.[5]
- BN: Bayesian Network for Image Classification method.
- RBN: Optimized Bayesian Network for Image Classification method with Relation.

Table 4.1: Comparison of Classification Results by NBNN, LNBNN, BN and RBN by 20 Test Images

Method	Number of Images in Training Set	
	15	30
NBNN	0.4650	0.5206
LNBNN	0.4821	0.5620
BN	0.5016	0.5906
RBN	0.5971	0.7231

Tab. 4.1 shows classification accuracy of different tested methods with 20 test images. Our optimized Bayesian Network for image classification method (RBN) shows the highest accuracy, 72.31%, among all the tested methods. This accuracy is much higher, more than 19% relatively higher, than those of the original NBNN. Compared with LNBNN, our method achieves more than 15% higher accuracy.

In Fig.4.1, it shows the classification accuracy in the listed classes with 15, 30 and 40 training images by 20 test images. we can see even the LNBNN performs better than NBNN in some classes but it also shows worse accuracy in a few classes. And Our method RBN performs better than both NBNN and LNBNN. And in some classes, the accuracy increase a lot, we can consider that among local features there are more dependence, for example, class Face, class Sunflower, class airplane, etc..

Furthermore, considerring the time consuming, LNBNN is the fastest method over all, however it cannot ensure the high quality of classification as shown in Fig. 4.1. Because the time complexity of our method is similar to NBNN, we gain the classification result for each query in the similar elapsed time, around 2.5s for the query image with 1000 patches. Additionally, BN is hundreds of times slower than RBN.

In conclusion, as shown in Tab.4.1 and Fig.4.1, our method achieves the most accurate classification results among all the tested methods. This is mainly because we consider the dependence among local features and the relation between highg-level feature and low-level features, instead of assuming all the local features are conditional independent.

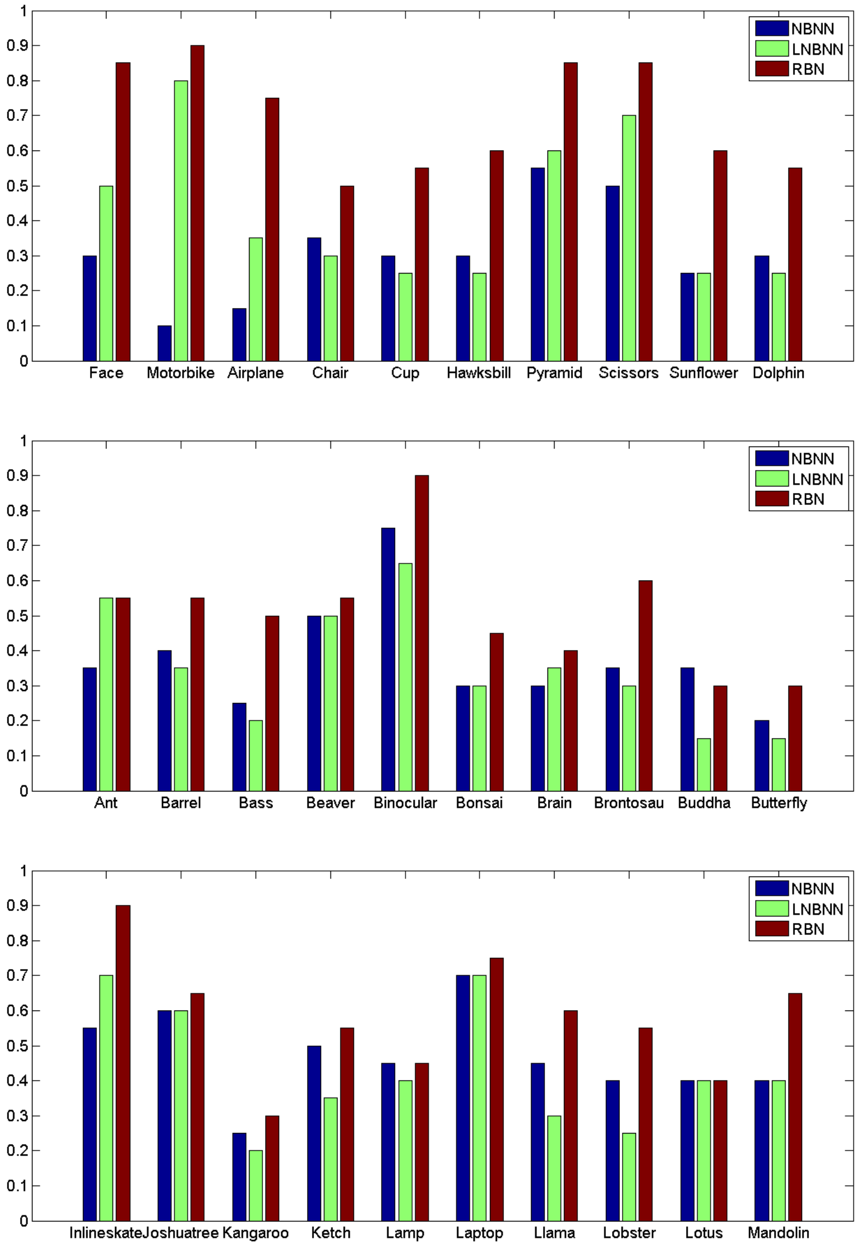

Figure 4.1: Comparison of classification results by NBNN, LNBNN and RBN in the training set of 15 images for each class.

5. Conclusion and Future Work

We have proposed Bayesian Network for image classifcation, a novel method defining and considering the high-level and low-level features and dependences among features for classification. Our method keeps the simplity of NBNN and estimates the relation of low level features for each high-level feature. With the layered structure of high-level and low-level features, we show the dependence among local features and build the Bayesian Network for image classification. By this method, we found that time complexity is too big and the result didn't come as well as expected. Then , we analyze the high-level and low-level features and define the relation between them. By this relation, we optimize our method with the faster speed and much higher accuracy. The experimental results in the Dataset Caltech101 show that our method outperforms around 20% over NBNN and LNBNN. And the time consuming is almost the same as NBNN for each query image.

As stated above, we already achieved quite improvement by this Bayesian Network. But we do believe that we can utilize more dependences among local features. And, because LNBNN is about hundreds of times faster than NBNN, so as our mehod. Therefore, for the further works, we should consider the time consuming to speed it up. Meanwhile, we should keep the classification results in the high quality.

요 약 문

이미지 분류를 위한 베이지안 네트워크

국소 특징들이 서로 조건부 독립이라는 가정을 갖고 있는 나이브 베이지안 근접 이웃 분류 (Naive Bayes Nearest Neighbor) 알고리즘은 학습 혹은 트레이닝 단계를 거치지 않고 직접적으로 분류를 할 수 있다는 장점을 갖고 있다. 실제로도 NBNN 계열의 알고리즘들은 기존의 비 매개 변수 분류기법 들보다 우수한 성능을 가질 수 있음이 입증되었다. 하지만 국소 특징 간의 조건부 독립이라는 가정은 실제와 맞지 않다는 약점, 즉 특징 간의 중요한 상관 관계를 놓칠 가능성을 안고 있다. 때문에 이 부분이 NBNN의 성능의 제약을 가져오는 단점을 해결하기 위해 본 연구에서는 이미지 분류를 위한 새로운 베이지안 네트웍 기반의 알고리즘을 제안하고 그 결과를 검증하였다. 정확도의 향상을 위해 국소 특징들간의 연관 관계를 패치의 크기와 이미지 상의 위치를 고려하여 고/저 차원으로 나누어 분석하였으며, 이를 통해 특징들 간의 연관성을 추출하는 과정에서 베이지안 네트웍의 최적화를 수행하였다. Caltech101 데이터셋을 통해 검증한 본 알고리즘은 기존 알고리즘들에 비해 최대 20% 까지의 정확도 향상을 보였으며 동시에 계산 시간 면에서는 큰 차이를 보이지 않음을 확인하였다.

References

[1] R. Behmo, P. Marcombes, A. Dalalyan, and V. Prinet. Towards optimal naive bayes nearest neighbor. In ECCV, 2010.

[2] O. Boiman and E. S. andM. Irani. In defense of nearestneighbor based image classification. In CVPR, 2008.

[3] F. Jurie and B. Triggs. Creating efficient codebooks for visual recognition. In ICCV, 2005.

[4] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In CVPR, 2006.

[5] S. McCann and D. G. Lowe. Local naive bayes nearest neighbor for image classification. In CVPR, 2012.

[6] T. Tuytelaars, M. Fritz, K. Saenko, and T. Darrell. The nbnn kernel. In ICCV, 2011.

[7] Z. Wang, S. Gao, and L.-T. Chia. Learning class-to-image distance via large margin and l1-norm regularization. In ECCV, 2012.

[8] Z.Wang, Y. Hu, and L.-T. Chia. Image-to-class distance metric learning for image classification. In ECCV, pages 706–719, 2010.

[9] Z. Wang, Y. Hu, and L.-T. Chia. Improved learning of i2c distance and accelerating the neighborhood search for image classification. In Pattern Recognition, 2011.

[10] Deng, J., Dong, W., Socher, R., Li, L., Li, K., and Fei-Fei, L. ImageNet: A large-scale hierarchical image database. Proc. CVPR 2009, IEEE (2009), 248–255

[11] T. Tommasi and B. Caputo, Frustratingly easy nbnn domain adaptation, in ICCV, 2013.

[12] David G. Lowe, Distinctive image features from scale-invariant keypoints, In International Journal of Computer Vision, 60, 2 (2004), pp. 91-110.

[13] David Heckerman, A Tutorial on Learning With Bayesian Networks, In Microsoft Research, 1995.

[14] H.Zhang, A.Berg, M.Maire, and J. Malik. Svm-knn: discriminative nearest neighbor classification for visual category recognition. In CVPR, 2006.

[15] S.Arya and H. Y. A. Fu. Expected-case complexity of approximate nearest neighbor searching. In Symposium on Discrete Algorithms, 2000.

[16] H. Zhang, The optimality of naive bayes, in Proceedings of the Seventeenth International Florida Artificial Intelligence Research Society Conference , (V. Barr and Z. Markov, eds.), Miami Beach, FL: AAAI Press, 2004

[17] Lewis, D. D.. Naive (Bayes) at forty: The independence assumption in information retrieval. In ECML, 1998.

[18] A. Bosch, A. Zisserman, and X. Munoz. Image classifcation using random forests and ferns. In ICCV, 2007.

[19] Neapolitan, Richard E. (2004). Learning Bayesian networks. Prentice Hall. ISBN 978-0-13-012534-7.

[20] Geiger, Dan; Verma, Thomas; Pearl, Judea. Identifying independence in Bayesian Networks. In Networks 20: 507–534, 1990.

[21] J. Pang, Q. Huang, B. Yin, L. Qin, and Q. Wang. Stochastic Boosting for large-scale image classification. In ICIP, 2013.

[22] R. Fei-Fei, L.and Fergus and P. Perona. Learning generative visual models from few training examples: an incremental bayesian approach tested on 101 object categories. In CVPR Workshop on Generative-Model Based Vision, 2004.

[23] M. Muja and D. Lowe. Fast approximate nearest neighbors with automatic algorithm conguration. In VISSAPP, 2009.

감 사 의 글

석사기간 동안 부족한 저를 이끌어 주시고 졸업할 수 있도록 많은 도움과 지도를 해주신 윤성의 교수님과 허재필님에게 먼저 감사드립니다. 교수님께서 많이 배려해주시고 챙겨주셔서 제가 졸업 논문을 낼 수 있었다고 생각합니다. 또한 함께 연구하고 생활을 한 SGLAB 연구실 동료들(Pio Caludio, 김동혁, 이가연, 김수민, 손명배, 최창민, 윤웅직, 권용선, 양현철, 김태영, 이윤석, Pend Du)에게도 감사드립니다. 한국 생활에 잘 적응할 수 있도록 교수님과 연구실 동료들에게 많은 도움을 받았던 점 감사드리고, 덕분에 석사기간 동안 많은 것을 배우고 갑니다. 마지막으로, 항상 저를 사랑해주시는 아버님에게 감사를 드리고 싶습니다.

이 력 서

이 름 : 순 밍 양

생 년 월 일 : 1988년 4월 28일

E-mail 주 소 : mingyang.sun@kaist.ac.kr

학 력

2004. 8. – 2007. 7. Jilin No.1 High School

2007. 8. – 2011. 7. Dalian University of Technology (B.S.)